|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Dùng làm gì | Dùng khi nào | Loại |
| Adapter | Adapter Pattern cho phép các inteface (giao diện) không liên quan tới nhau có thể làm việc cùng nhau. Đối tượng giúp kết nối các interface gọi là Adapter. | Gặp 1 interface k phù hợp lớp đang viết  K thể kế thừa A nhưng cần 1 lớp B có các chức năng như A  Cần phải có sự chuyển đổi interface từ nhiều nguồn khác nhau |  |
| Strategy | Nó cho phép định nghĩa tập hợp các thuật thoán , hành vi thành lớp độc lập và dễ dàng thay đổi linh hoạt các thuật toán bên trong object . do đó dễ dàng phát triển | Khi muốn thay đổi các thuật toán được sử dụng bên trong một đối tượng tại thời điểm run-time  Khi có 1 đoạn mã muốn phát triển cần tách ra khỏi phần chính |  |
| Prototype | Tạo ra một khuôn mẫu . Và sử dụng hàm clone để cache 1 bản sao trên sử dụng nó để sử dụng cho các dữ liệu tiếp theo và lưu cơ sở dữ liệu khi nào cẩn do đó giảm các cuộc gọi cơ sở dữ liệu làm tăng hiệu năng | Sử dụng khi có quá nhiều request đến server làm giảm bớt thời gian xử lý , giảm bớt lượng object tạo ra trong cùng 1 thời điểm, |  |
| Flyweight | Dùng để tái sử dụng đối tượng tương tự đã tồn tại bằng cách lưu trữ chúng hoặc tạo đối tượng mới néu k tìm thấy đối tượng phù hợp | Khi có một số lượng lớn các đối tượng được ứng dụng tạo ra một cách lặp đi lặp lại  Khi tạo đòi hỏi nhiều bộ nhớ và thời gian  Khi muốn tái sử dụng  Khi nhóm đối tượng chứa nhiều đối tương tương tự và 2 đối tượng trong nhóm k khác nhau nhiều |  |
| State | Là 1 bản nâng cấp của Strategy Pattern  Nó cho phép một đối tượng thay đổi hành vi của nó khi trạng thái nội bộ của nó thay đổi. Đối tượng sẽ xuất hiện để thay đổi lớp của nó. | Sử dụng khi hành vi của object phụ thuộc vào trạng thái của nó  Khi nhiều điều kiện phức tạp buộc đối tượng phụ thuộc vào trạng thái của nó |  |